98P1600 REEAL

(12) UK Patent Application (19) GB (11) 2 189 061 (13) A

B1

(43) Application published 14 Oct 1987

(21) Application No 8703190

(22) Date of filing 12 Feb 1987

(30) Priority data
(31) 61/051669 (32) 10 Mar 1986 (33) JP
61/101539 1 May 1986

(51) INT CL⁴
G06F 15/16

(52) Domestic classification (Edition I):
G4A MP

(56) Documents cited
GB 1556228 US 4472771 US 4075693

(58) Field of search
G4A
Selected US specifications from IPC sub-class G06F

(71) Applicants
Hitachi Ltd.

(Incorporated in Japan)

6 Kanda Surugadai 4-chome, Chiyoda-ku, Tokyo, Japan.

Hitachi Software Engineering Co. Ltd.

(Incorporated in Japan)

91 Onoecho-6-chome, Naka-ku, Yokohama, Japan

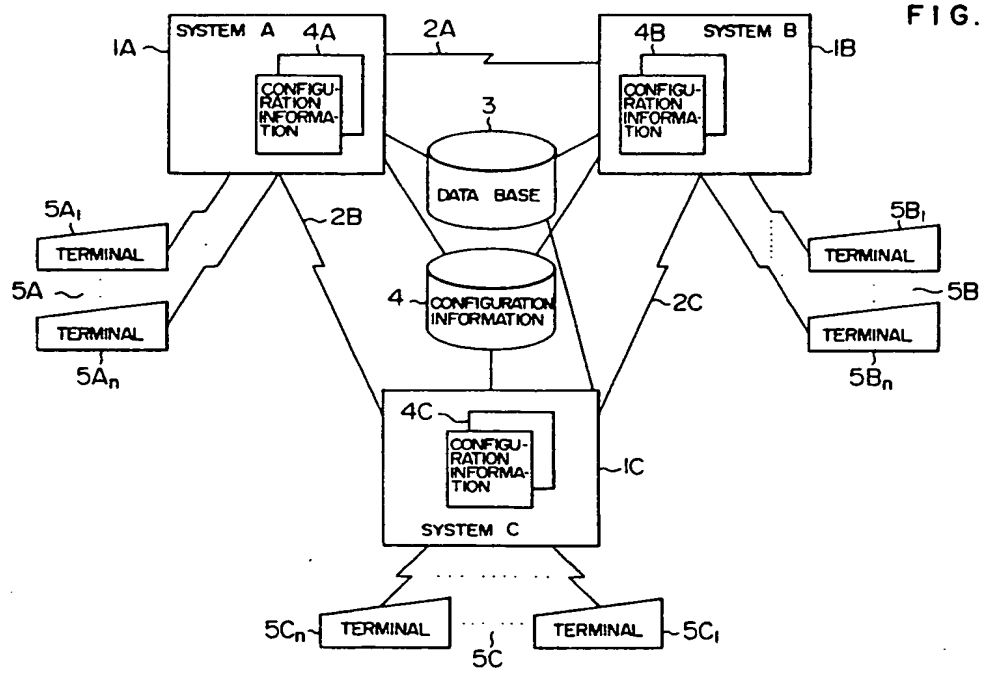
(72) Inventors
Hiromichi Ogata
Tetsuo Murakawa
Kengo Ogawara

(74) Agent and/or Address for Service
J. A. Kemp & Co.,
14 South Square, Gray's Inn, London WC1R 5EU

(54) Management of system configuration data

(57) In a computer system having a plurality of data processing systems (A, B, C), a communication device (2A, 2B, 2C) by which the data processing systems exchange information, and a data base (3) shared by the data processing systems and independently accessible by each, the configuration information of each data processing system is collectively stored in a common store device (4) commonly accessible by each data processing system, similarly to the shared data base, and each data processing system receives the configuration information directly from the common store device. A data processing system issuing a configuration change request executes a change to the configuration information stored in the store device, after it receives permission messages from the other data processing systems via the communication device.

FIG. 1



GB 2 189 061 A

FIG. 1

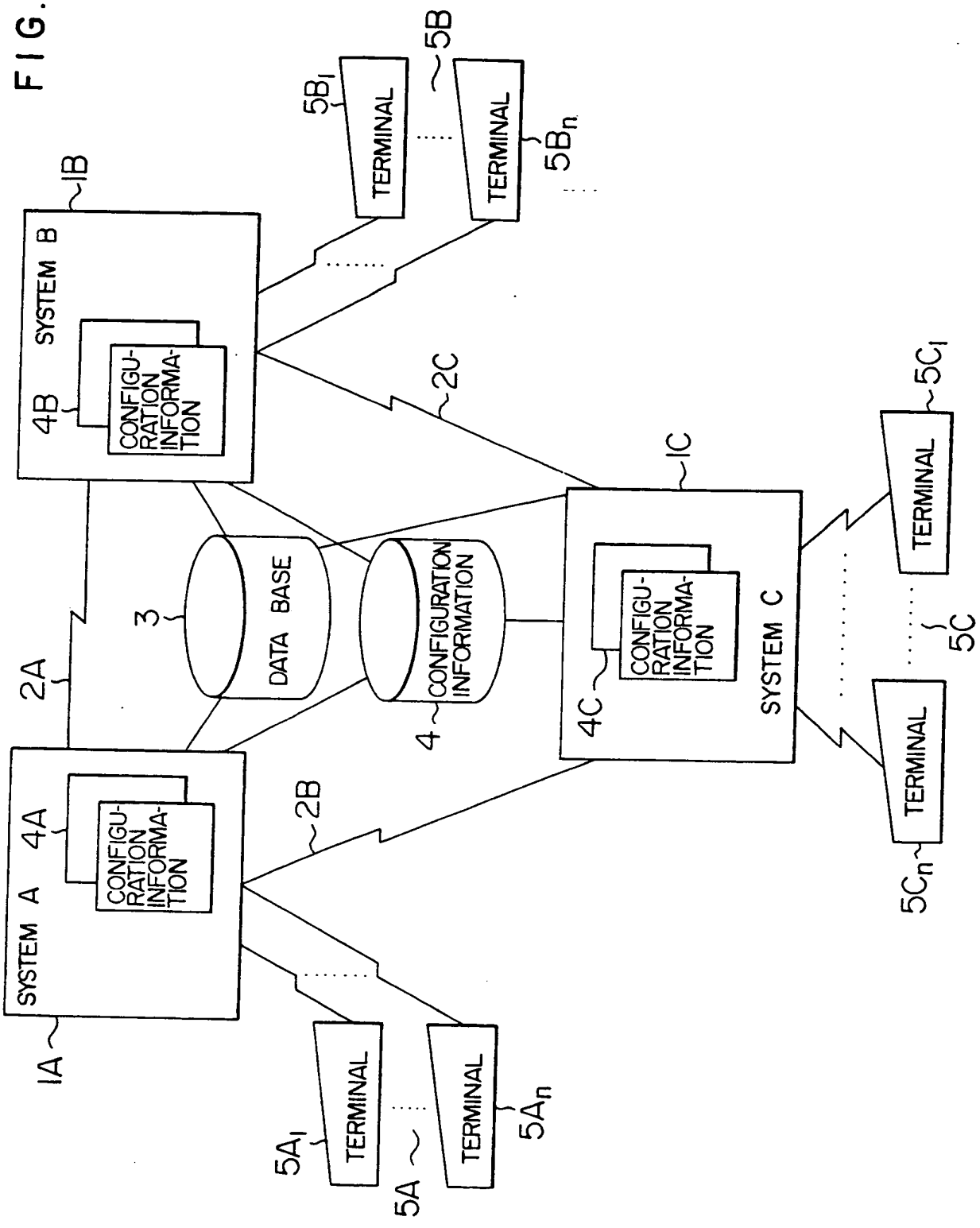

SYSTEM A
SYSTEM B
SYSTEM C
CONFIGU-RATION INFORMA-TION
DATA BASE
CONFIGURATION INFORMATION
TERMINAL
1A
1B
1C
2A
2B
2C
3
4
4A
4B
4C
5A
5B
5C
5A1
5An
5B1
5Bn
5C1
5Cn

FIG. 2

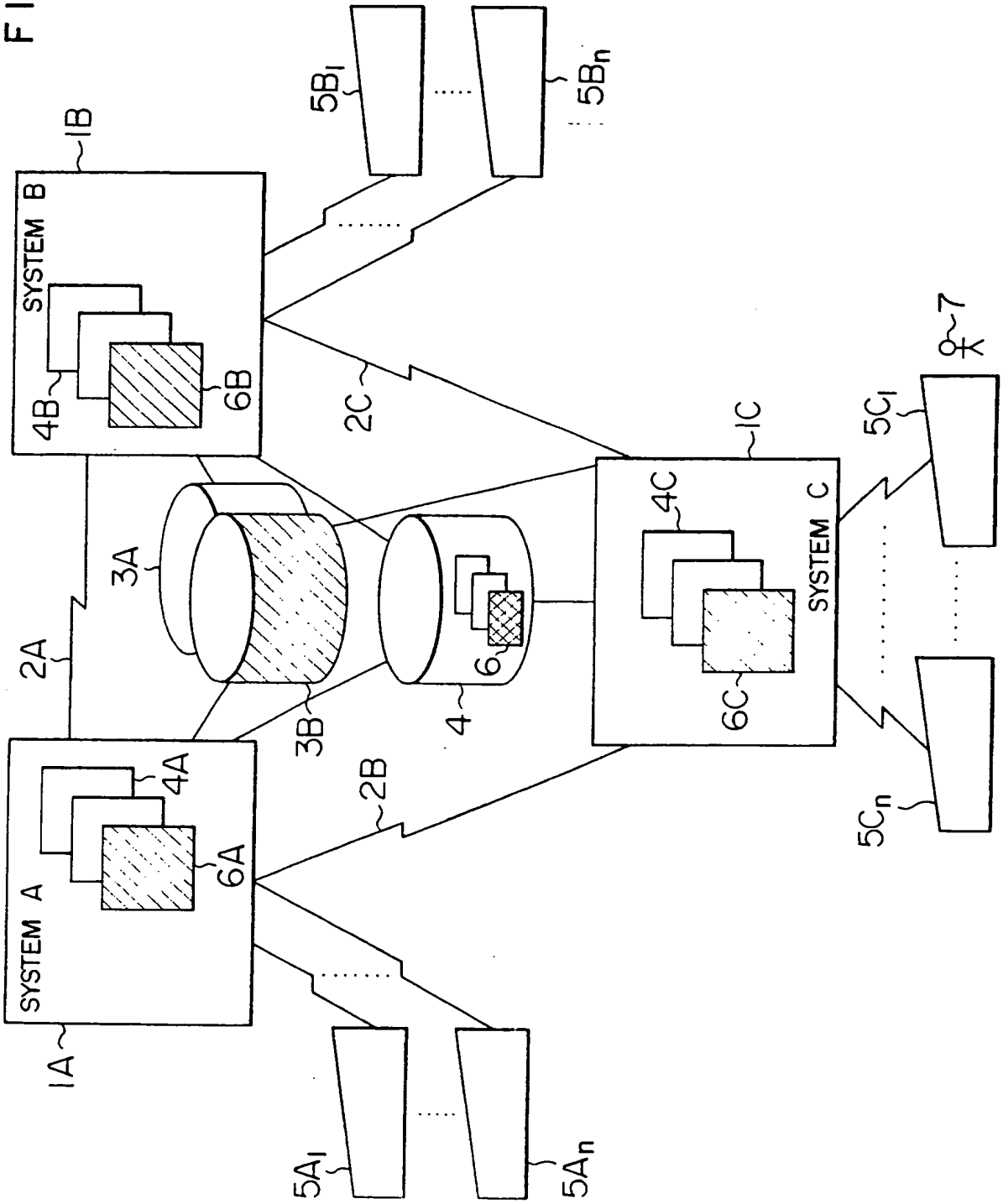

1A
1B
1C
SYSTEM A
SYSTEM B
SYSTEM C
2A
2B
2C
3A
3B
4
4A
4B
4C
5A1
5An
5B1
5Bn
5C1
5Cn
6A
6B
6C
6
7

FIG. 3

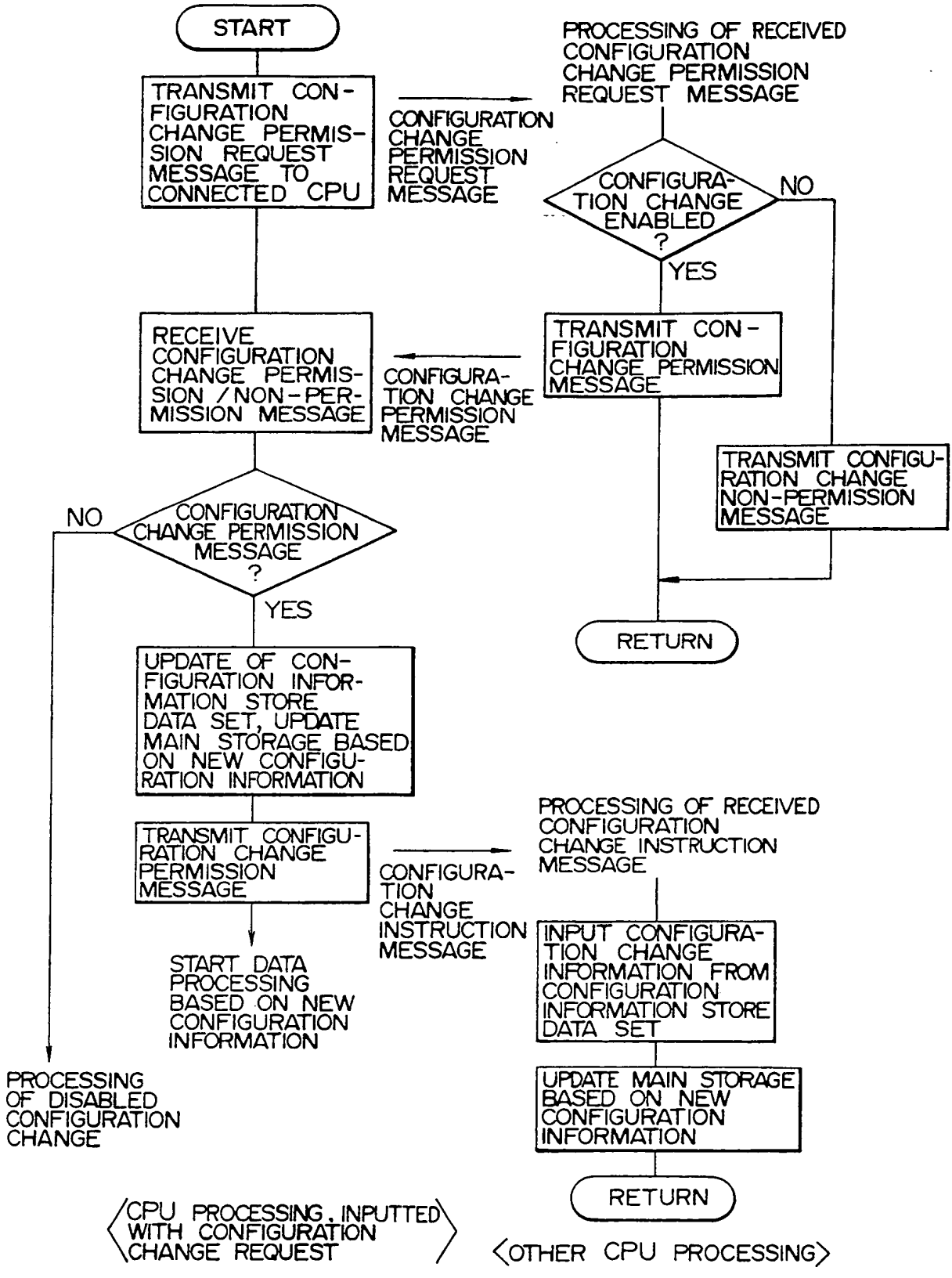

START
TRANSMIT CONFIGURATION CHANGE PERMISSION REQUEST MESSAGE TO CONNECTED CPU
CONFIGURATION CHANGE PERMISSION REQUEST MESSAGE
PROCESSING OF RECEIVED CONFIGURATION CHANGE PERMISSION REQUEST MESSAGE
CONFIGURATION CHANGE ENABLED ?
NO
YES
TRANSMIT CONFIGURATION CHANGE PERMISSION MESSAGE
CONFIGURATION CHANGE PERMISSION MESSAGE
RECEIVE CONFIGURATION CHANGE PERMISSION / NON-PERMISSION MESSAGE
TRANSMIT CONFIGURATION CHANGE NON-PERMISSION MESSAGE
RETURN
CONFIGURATION CHANGE PERMISSION MESSAGE ?
NO
YES
UPDATE OF CONFIGURATION INFORMATION STORE DATA SET, UPDATE MAIN STORAGE BASED ON NEW CONFIGURATION INFORMATION
TRANSMIT CONFIGURATION CHANGE PERMISSION MESSAGE
CONFIGURATION CHANGE INSTRUCTION MESSAGE
PROCESSING OF RECEIVED CONFIGURATION CHANGE INSTRUCTION MESSAGE
START DATA PROCESSING BASED ON NEW CONFIGURATION INFORMATION
INPUT CONFIGURATION CHANGE INFORMATION FROM CONFIGURATION INFORMATION STORE DATA SET
PROCESSING OF DISABLED CONFIGURATION CHANGE
UPDATE MAIN STORAGE BASED ON NEW CONFIGURATION INFORMATION
RETURN
〈CPU PROCESSING, INPUTTED WITH CONFIGURATION CHANGE REQUEST〉
〈OTHER CPU PROCESSING〉

FIG. 4

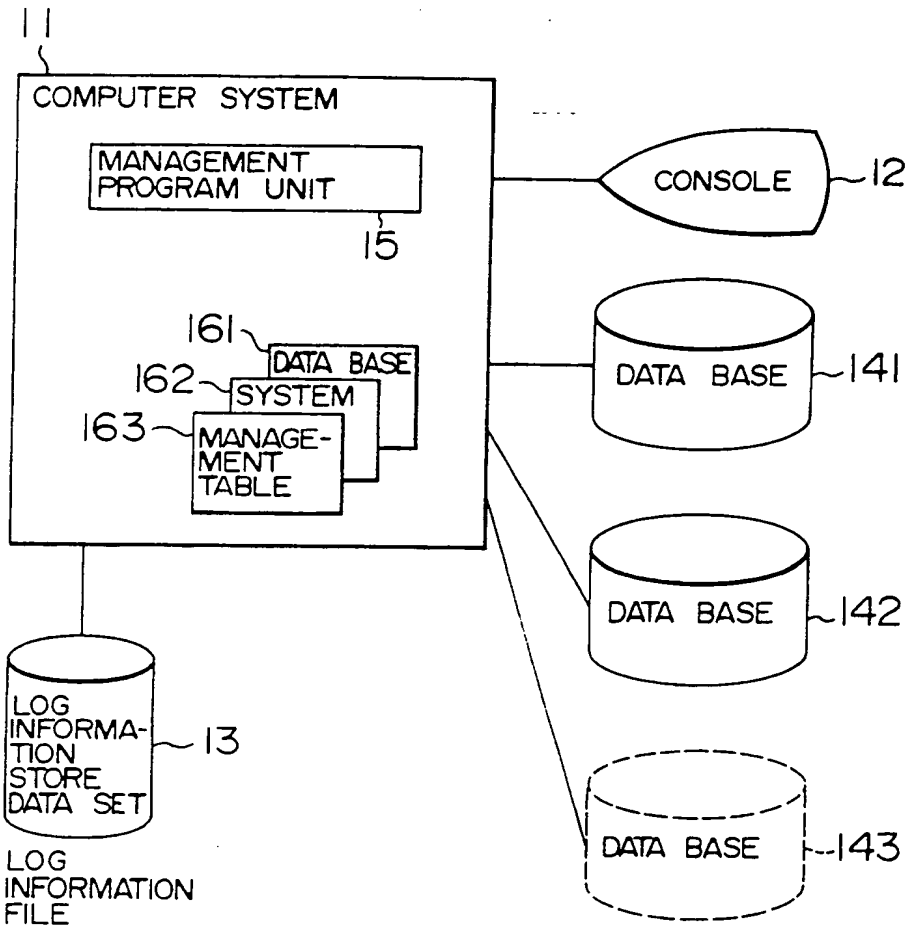


FIG. 7

CONFIGURATION CHANGE REQUEST COMMAND

CHANGE EVENT	NAME	ATTRIBUTE

FIG. 5

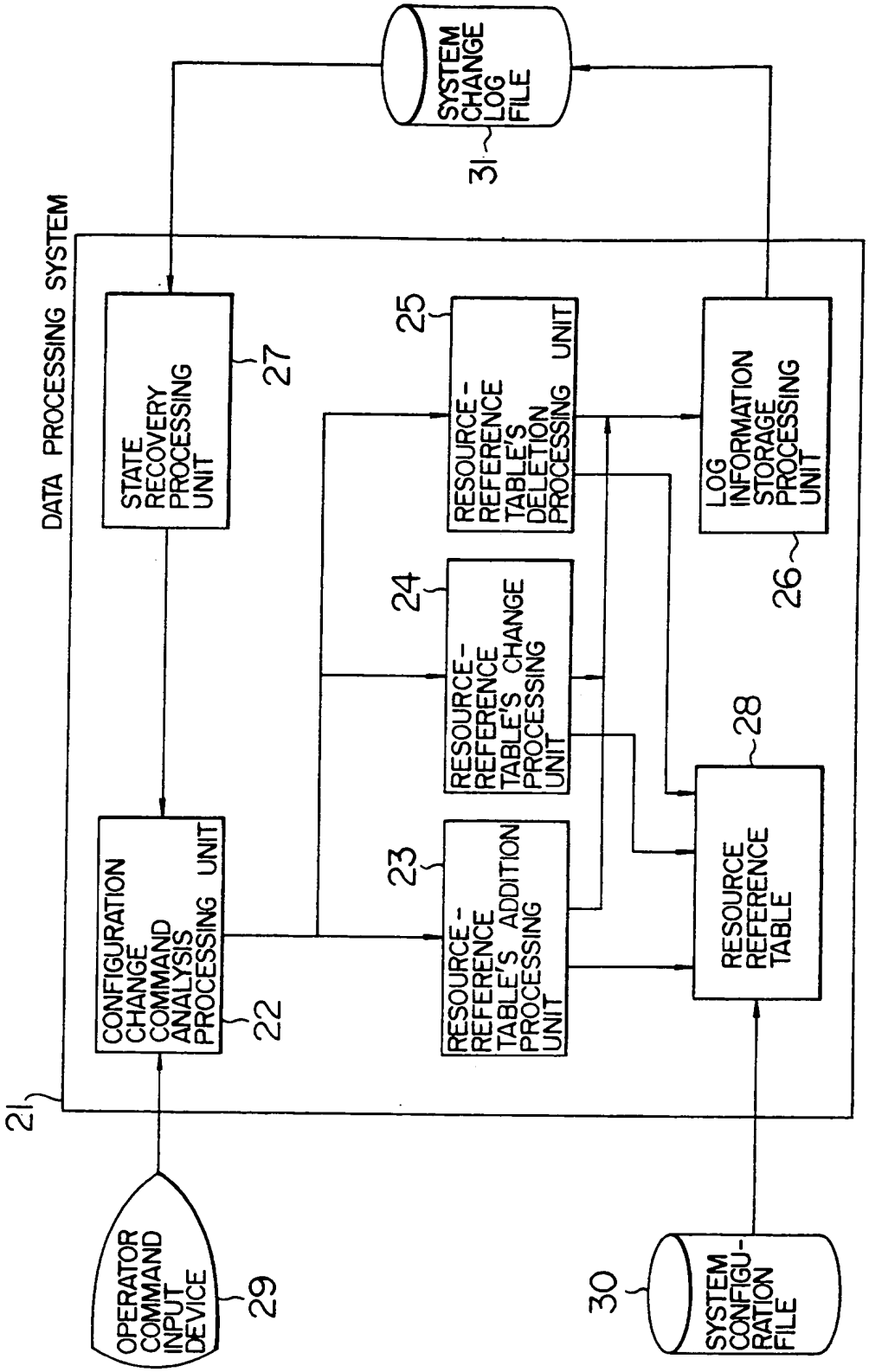

SYSTEM CHANGE LOG FILE
31
DATA PROCESSING SYSTEM
STATE RECOVERY PROCESSING UNIT
27
RESOURCE-REFERENCE TABLE'S DELETION PROCESSING UNIT
25
LOG INFORMATION STORAGE PROCESSING UNIT
26
RESOURCE-REFERENCE TABLE'S CHANGE PROCESSING UNIT
24
RESOURCE REFERENCE TABLE
28
CONFIGURATION CHANGE COMMAND ANALYSIS PROCESSING UNIT
22
RESOURCE-REFERENCE TABLE'S ADDITION PROCESSING UNIT
23
21
OPERATOR COMMAND INPUT DEVICE
29
SYSTEM CONFIGU-RATION FILE
30

FIG. 6

TERMINAL REFERENCE TABLE

81

TERMINAL NAME	TERMINAL ATTRIBUTE			VALIDITY
	TERMINAL TYPE	CONNECTION LINE TYPE	STATUS	

LINE REFERENCE TABLE

82

LINE NAME	LINE ATTRIBUTE			VALIDITY
	LINE TYPE	CONNECTED TERMINAL ADDRESS	STATUS	

DATA BASE REFERENCE TABLE

83

DATA BASE NAME	DATA BASE ATTRIBUTE			VALIDITY
	DATA BASE TYPE		STATUS	

SPECIFICATION

Configuration information management method and apparatus for a computer system

BACKGROUND OF THE INVENTION

The present invention relates to a management method and apparatus of configuration information on data processing system and data base, in a computer system such as a banking system which has a plurality of information processing systems and a common data base accessible independently by the information processing systems.

Recently, there has been a strong need of a computer system in which information can be exchanged among a plurality of data processing systems and each system can independently access a common data base in association with other data processing systems. As the system configuration information, configuration information or structure information on data processing systems and configuration information on a data base are required, the former information including types of data processing system and disk device, presence/absence of communication means, communication protocol, destination system ID, and so on, and the latter information including type of data set, name of data set, type of disk device, details of disk area, for example. Conventionally, such system configuration information has been managed, however, independently in each data processing system.

A computer system wherein a plurality of data processing systems can access a common data base is disclosed, for example, in JP-A-59-2163 Laid Open July 1, 1984. The system processes a measure to a trouble or failure (CPU-down) of each data processing system, but management of system configuration information is not considered.

In a prior art, each data processing system independently manages the system configuration information. Therefore, when the system configuration information is to be changed, all of the data processing systems must change their system configuration information at a time, thus increasing the amount of change works in proportion to the number of data processing systems. In addition, disagreement of system configuration information between data processing systems is likely to occur. For instance, there are some cases where the data base cannot be accessed or a communication between systems is not available, because of disagreement of data base configuration information or disagreement of definition of communication means, between data processing systems.

Apart from the above, in a long-run on-line system, addition, deletion or the like of the resources such as a data base, terminals and the like, may become necessary during running. However, for a case that the system configuration is changed during running and that the system stops due to some causes such as a failure, a conventional system has not given much consideration to automatically recover the state after change of the configuration (the state before stopping of the system such as system-down). As a result, the operator must handle such a case to recover the system, which takes a long time and results in an interruption of application transaction, particularly of on-line application in an on-line system.

Change of on-line system configuration is discussed in "Latest Data Base System and Applications" in a separate volume of Japanese literature "Bit" (published by Kyoritsu Publication Co., on January 10, 1984), page 53.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide a configuration information management method and an apparatus therefor capable of collectively and highly reliably managing configuration information on a plurality of data processing systems, in a computer system wherein a plurality of data processing systems can exchange information using communication means and each system can independently access a common data base.

It is another object of the present invention to provide a configuration information management method and an apparatus therefor as above capable of reducing a configuration information change work and preventing an error in change work.

It is a further object of the present invention to provide a configuration information management method and an apparatus therefor capable of reducing an overhead of an automatic recovery processing for a case that the system configuration is changed during running and the computer system is thereafter stopped.

According to one aspect of the present invention, in a computer system wherein a plurality of data processing systems can exchange information and each data processing system can directly access a common data base, a memory device is provided which can be commonly or non-exclusively accessed by each data processing system, and system configuration information is collectively stored in the memory device for management thereof. With such a construction, each data processing system fetches its configuration information from the commonly accessible memory device. The system configuration information is changed by changing the content of the commonly accessible memory device. At that time, the data processing system changing its system configuration information informs the other data processing systems of such effect, using system communication

means. Thus, each data processing system can concurrently refer to common configuration information so that contradiction and inconsistency in configuration information among individual data processing systems can be avoided. In addition, configuration information on each of a plurality of data processing systems can be collectively changed in the common memory device so that the configuration information change work can be reduced and an error in the work can be prevented. Further, adding information to the data base can be effected without suspending service of terminal systems, thus improving the availability of an on-line system.

According to another aspect of the present invention, a system change log file using a magnetic disk for example is provided, which file stores system configuration change log during running of the system. In operation, system configuration change is requested using an operator command when addition, deletion, change or the like with respect to the system configuration becomes necessary while running the system. After processing this request, the command is stored in the system change log file. When it becomes necessary to recover the system after it is stopped due to some causes, the contents of system configuration change during running of the system are sequentially re-executed based on commands stored in the system change log file, to thereby recover the same system configuration as that when the system last stopped. Therefore, an overhead of recovery processing can be reduced and the system availability can be improved.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows the whole system arrangement of an embodiment according to the present invention.

Fig. 2 is a diagrammatical view explaining the operation of changing system configuration information according to the present invention.

Fig. 3 is a flow chart for explaining the operation of changing system configuration information, in connection with Fig. 2.

Fig. 4 shows an example of a data processing system which changes the configuration information during on-line operation.

Fig. 5 is a functional block diagram showing another embodiment of the present invention.

Fig. 6 shows an example of a recovery reference table.

Fig. 7 shows an example of the format of a configuration change request command.

DESCRIPTION OF THE PREFERRED EMBODIMENT

An embodiment of the present invention will be described with reference to the accompanying drawings.

Fig. 1 shows the overall arrangement of an embodiment according to the present invention. In the computer system shown in Fig. 1, three data processing systems 1A, 1B and 1C exchange information via communication lines 2A, 2B and 2C, and each data processing system independently and directly accesses a common data base 3 which is usually in a hardware form of such as disk/direct-access-storage device. Configuration information on the data processing systems 1A, 1B and 1C is stored in a memory device 4 (magnetic disk device or semiconductor memory device) commonly accessible by the data processing systems. A part of the configuration information is transferred to a configuration information store area 4A, 4B or 4C of each data processing system from the memory device 4, when the system starts or when the configuration information in the memory device is updated by any one of the data processing systems requesting configuration information change. Each data processing system 1A, 1B and 1C is connected with a plurality of terminal systems $5A_1$, $5A_2$, ..., $5B_1$, $5B_2$, ..., $5C_1$, $5C_2$, ... Upon reception of a message from one of the terminal systems 5A, 5B and 5C, the data processing system 1A, 1b or 1C concerned informs the other data processing systems over the communication lines 2A, 2B and 2C of whether it is allowed to access the data base 3 or not. Thus, each data processing system can directly access the data base 3, without requiring full management by the other data processing systems. Such a direct access method to a common data base can generally conceptually employ the technique disclosed in the above-referenced related application. The construction for managing the configuration information at each data processing system is exemplified in Fig. 4.

Next, the operation of altering system configuration information will be described with reference to Figs. 2 and 3.

Assume now that the data processing system 1C adds a new data base 3B to the data base 3A commonly used by the data processing systems (abbreviated as CPU) 1A, 1B and 1C. In this case, an operator 7 inputs new or additional configuration information 6 (such as data set type, data set name and the like of the data base 3B) through the terminal system $5C_1$ to the memory device 4. Upon this inputting of additional configuration information, the data processing system 1C transmits a change permission request message to the data processing systems 1A and 1B via the communication lines 2B and 2C. Upon reception of this message, the data processing systems 1A and 1B check the vacant state of their main storages or check if the additional configuration information duplicates the resources other than that commonly used. If there is no problem, the data processing systems 1A and 1B transmit change permission information to the data processing system 1C, and at the same time, inhibit by themselves an access to the

configuration information memory device 4. For instance, the disk controller in the data processing system 1C issues a command to configuration information file disks (volumes) of the other processing systems to inhibit an access. In such a case, each data processing system is not required to suspend its service to the terminal systems, but a request for changing the data base and/or configuration information may be further accepted while giving such service. Upon reception of the information change permission from the data processing systems 1A and 1B, the data processing system 1C stores the additional configuration information 6 in the common configuration information memory device 4. After completion of storage, change completion is informed to the data processing systems 1A and 1B via the communication lines 2B and 2C to release the self-inhibition of access of the systems 1A and 1B to the memory device 4. Thereafter, the data processing systems 1A, 1B and 1C can independently access the memory device 4 to add the same contents as of the additional configuration information in their own main storage at areas 6A, 6B and 6C. At the same time, pointers to the store areas 6A, 6B and 6C of the additional configuration information are given so that they can be pointed out while referring to the store areas 4A, 4B and 4C using a reference table therein. Fig. 3 shows the flow chart of the configuration change processing.

With the method wherein the data processing system 1C communicates the additional configuration information with the data processing systems 1A and 1B without using the common configuration information memory device 4, it is necessary to communicate a large amount of configuration information with the systems 1A and 1B via the communication lines 2B and 2C, as compared with this embodiment. This results in a poor performance of executing an exchange of exclusive control information of the data base via the communication line, and in a poor performance of executing a transfer of information between data processing systems. Further, in recovering the configuration information of a data processing system in failure, it is necessary to receive the additional configuration information of the other data processing systems to ensure agreement of configuration information among the systems. This results in an increase of a system recovery processing time, and hence poor reliability and performance.

Next, an example of change management of configuration information will be described referring to Fig. 4.

In Fig. 4, a computer system 11 includes therein a management program unit 15, and data base management tables 161, 162 and 163, and is connected outside of the system to a console 12, a log information store data set 13, and data base units 141 to 142. The data base management tables 161 and 162 manage the data base units 141 and 142, while the data base management table 163 is a table pr pared for use in adding a new data base unit 143. In adding the new data base unit 143, the operator supplies the name and location of the data base unit to be added to the management program unit 15 via the console 12. The management program unit 15 checks if the additional data base unit can be added or not, upon confirming if a reserved, additional table is present among the data base management tables 161 to 163 or if the name is the same as those of the other data base units 141 and 142. If possible, the management program unit 15 operates to make the log information store data set 13 store the configuration change information, and also to make the added data base management table 163 store the data of name and location of the data base unit 143. As above, any desired data processing system can change the configuration information to perform addition/change of a data base unit.

Next, an embodiment of change management of configuration information will be described with reference to Figs. 5 to 7.

Fig. 5 is a functional block diagram of an embodiment of the present invention. In Fig. 5, a data processing system 21 has, as elements relevant to the present invention, a configuration change command analysis unit 22, a resource-reference table's addition processing unit 23, a resource-reference table's change processing unit 24, a resource-reference table's deletion processing unit 25, a log information storage processing unit 26, a state recovery processing unit 27 for system recovery, and a resource reference table 28. Connected to this data processing system 21 are an operator command input device 29, a system configuration file 30, and a system change log file 31. The system configuration file 30 and the system change log file 31 use an external memory device such as a magnetic disk.

The data processing system 21 first loads the system configuration information in the system configuration file 30 at the start of the system, to accordingly construct the system. Fig. 6 shows an example of the resource reference table 28, which includes a terminal reference table 81, a communication line reference table 82, and a data base referenc table, respectively corresponding to the terminals, communication lines, data bases connected to the data base processing system 21.

If addition, change, deletion or the like with respect to terminal, communication line, data base or the like becomes necessary during running of the data processing system, the operator inputs a configuration change request command via the operator command input device 29. The configuration change request

command is constructed, as shown in Fig. 7, of a configuration change event (addition, change, deletion or the like with respect to terminal, communication line, data base or the like), a name (of terminal, communication line, data base or the like concerned), and an attribute (e.g., terminal type, communication line type or the like). The configuration change command analysis processing unit 22 of the data processing unit 21 analyzes a configuration change request command inputted from the operator command input device 29 and thereafter, the control is assigned to the resource-reference table's addition processing unit 23, the resource-reference table's change processing unit 24, or the resource-reference table's deletion processing unit 25, respectively for addition, change or deletion. Particularly, in case of adding a terminal, the terminal name and the attribute in the configuration change request command are stored in the vacant area of the terminal reference table 81 under control of the resource-reference table's addition processing unit 23, to thereby establish a corresponding relation with the added resource. In case of changing a terminal, a part or all of the contents in the corresponding entry of the terminal reference table 81 are changed under control of the resource-reference table's change processing unit 24. In case of deleting a terminal, the corresponding entry of the terminal reference table is invalidated. Addition, change, or deletion for a communication line or a data base is conducted in a similar manner.

The resource-reference table's addition unit 23, the resource-reference table's change unit 24, and the resource-reference table's deletion unit 25 check the contacts of the resource reference table 28 and the processing results therein, are respectively informed to the log information storage processing unit 26. Based on the processing results, the log information storage processing unit 26 generates log information to sequentially store it in the system change log file 31. The log information to be stored in the system change log file 31 has basically the same format as of the configuration change request command shown in Fig. 7.

When the system is re-started after it stopped owing to some causes, the system configuration information in the system configuration file 30 is again loaded in the resource reference table 28. On the other hand, the state recovery processing unit 27 sequentially reads the log information (history information) in the system change log file 31 to send it to the configuration change command analysis processing unit 22 which, similar to the case of the operator command (configuration change request command) from the operator command input device 29, analyzes the log information and thereafter, the control is passed to the resource-reference table addition processing unit 23, the change processing unit 24 or the deletion processing unit 25. As described above, the contents of the system change log file 31 are processed at the state recovery processing unit 27, the configuration change command analysis processing unit 22, and the resource-reference table's addition/change/deletion processing units 23, 24 and 25, in this order. As a result, the contents of the resource reference table 28 recover the state when the system stopped, and hence the system recovers the same configuration when it stopped.

CLAIMS

1. In a computer system wherein a plurality of data processing systems exchange information and each data processing system directly accesses a common data base, a configuration information management method comprising the steps of:

(a) providing a common configuration information store memory directly accessible by each data processing system, and storing configuration information of each data processing system; and

(b) fetching the configuration information of each data processing system only from said common configuration store memory, and changing the configuration information in said store memory from any desired one of said data processing system when requested.

2. A method according to claim 1, wherein said fetching step (b) includes issuing a configuration information change permission request to the other data processing systems prior to execute the change, and executing the configuration information change in response to reception of permission messages from said other data processing systems.

3. A method according to claim 1, further including the steps of:

providing means for changing the system configuration during running of said computer in a software manner, providing a history information store memory, and storing system configuration information changed during running in said history information store memory; and

after occurrence of the system stop, recovering the system configuration by reproducing the configuration information when the system stopped, based on the contents stored in said history information store memory.

4. In a data processing system whose system configuration can be changed during its running, a method for managing data processing system configuration information comprising the steps of:

storing change history information of system configuration information changed during the running in a memory;

sequentially reading change history information in said memory in a recovery process after the system stopped due to a failure; and

recovering the system configuration information when the system stopped, based on the read change history information.

5. In a computer system wherein a plurality of data processing systems transfer information to each other and each data processing system independently accesses a common data base, an apparatus for managing computer system configuration information comprising:

store means for storing configuration information of each data processing system, said store means being directly and commonly accessed by each data processing system, in a similar manner to said common data base;

communication means for communicating the data processing systems to each other and for performing data transfer between said store means and each data processing system; and

change enabling means associated with said communication means for permitting each data processing means to directly change the configuration information stored in said store means.

6. The apparatus according to claim 5, wherein said change enabling means includes control means provided for each data processing system for managing the configuration information therefor by a program.

7. A computer system configuration information management method substantially as hereinbefore described with reference to the accompanying drawings.

Printed for Her Majesty's Stationery Office
by Burgess & Son (Abingdon) Ltd, Dd 8991685, 1987.
Published at The Patent Office, 25 Southampton Buildings,
London, WC2A 1AY, from which copies may be obtained.